Machine Learning (CS 7641 – Spring 2018)

Problem Set 2

GTID: pbharath6 17/04/2018

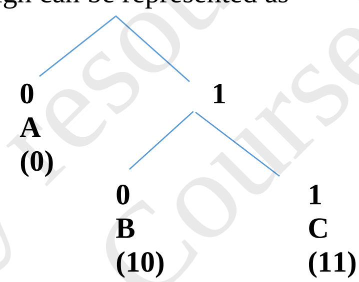
1. **You have to communicate a signal in a language that has 3 symbols A, B and C. The probability of observing A is 50% while that of observing B and C is 25% each. Design an appropriate encoding for this language. What is the entropy of this signal in bits?**

   To communicate a signal in a language that has 3 symbols with given probability
   A = 50%
   B = 25 %
   C = 25%
   We have to determine the Entropy of the signal to send the least number of bits required to transfer the information.
   The entropy of a symbol is given as Sum (Probability of the Symbol P(s) * size of the symbol (no of bits)

   E (number of bits per symbol) = P(s) * size of a symbol; size of a symbol = log 1/ P(s)

   Hence, E(S) = P(s) * log 1/ P(s)

   For the given symbols, the design can be represented as



Symbol	**Bits**
A	**0**
B	**10**
C	**11**

   Entropy of sending this signal = P (A) * no of bits (A) + P (B) * no of bits (B) + P (C) * no of bits (C)
   E (signal) = 0.5 * 1 + 0.25 * 2 + 0.25 * 2
   E (signal) = 0.5 + 0.5 + 0.5 = **1.5 bits**

2. **Show that the Kmeans procedure can be viewed as a special case of the EM algorithm applied to an appropriate mixture of Gaussian densities model.**

   Here we assume a model with parameters Θ defining probabilities of the form PΘ(x1, . . . , xN , z1, . . . , zN). In a mixture of Gaussian model we have that each xt is a point in RD and each zt is a class label with zt ∈ {1, . . . , K}. We now consider an arbitrary model defining PΘ(x1, . . . , xn, z1, . . . , zn). In unsupervised training we are given only x1, . . . , xn and, given only this information, we set the parameter values Θ of the model. Hard EM approximatey solves the following optimization problem. Θ ∗ = argmax Θ max z1,...,zn PΘ(x1, . . . , xn, z1, . . . , zn) (1)

K-means clustering is a special case of hard EM. In K-means clustering we consider sequences $\mathbf{x_1}$……………………$\mathbf{x_n}$ and $\mathbf{z_1}$…………… $\mathbf{z_n}$ with $\mathbf{x_t} \in \mathbf{R^D}$ and $\mathbf{z_t} \in \{1\ldots\ldots\ldots K\}$.
In other words, $\mathbf{z_t}$ is a class label, or cluster label, for the data point $\mathbf{x_t}$. We can define a K-means probability model as follows where **N (μ, I)** denotes the D-dimensional Gaussian distribution with mean $\mu \in \mathbf{R^D}$ and with the identity covariance matrix.

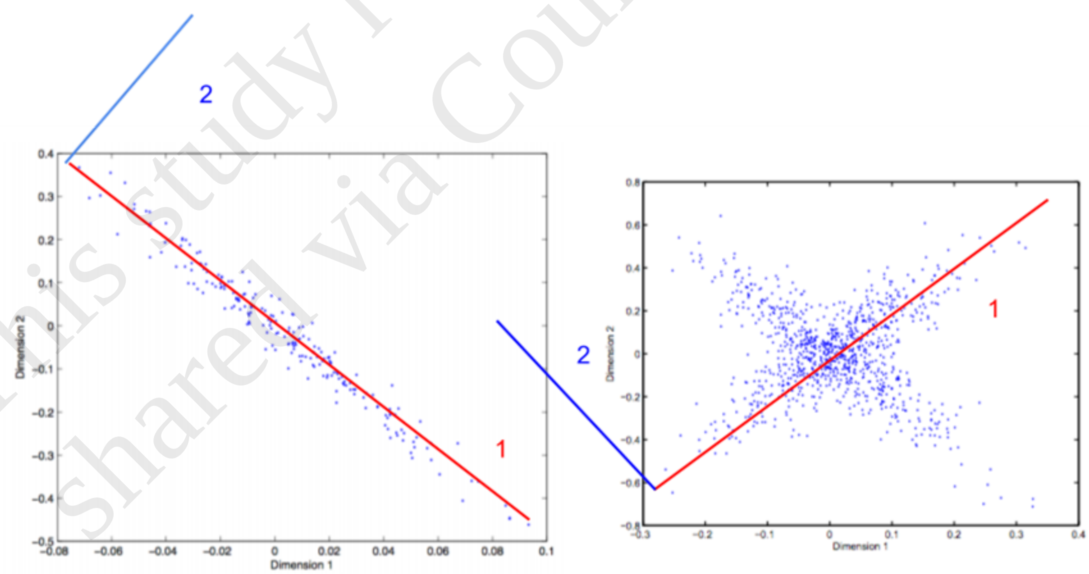
$\Theta = (\mu^1, \ldots, \mu^K), \mu^k \in R^D$

$P_\Theta(x_1 \ldots\ldots x_n, z_1 \ldots\ldots z_n)$

= PhI (t=1…N) ($P_\Theta(z_t)P_\Theta(x_t|z_t)$)
= PhI (t=1…N) (1/K N (μzt , I)(xt)

We now consider the optimization problem defined by (1) for this model. For this model one can show that (1) is equivalent to the following. (µ1 , . . . , µK) ∗ = argmin µ1,...,µk min z1,...,zn X N t=1 ||µ zt − xt||2 (2) The optimization problem (2) defines K-means clustering (under quadratic distortion). This problem is nonconvex and in fact is NP-hard (worse than nonconvex). The K means algorithm is coordinate descent applied to this objective and is equivalent to hard EM under tha above probability model. The K-means clustering algorithm can be written as follows where we specify a typical initialization step. 1. Initialize µ z to be equal to a randomly selected point xt. 2. Repeat the following until (z1, . . . zn) stops changing. (a) zt := argminz ||µ z − xt||2 (b) Nz := |{t : zt = z}| (c) µ z := 1 Nz P t: zt=z xt. In words, the K-means algorithm first assigns a class center µ z for each class z. It then repeatedly classifies each point xt as belonging to the class whose center is nearest xt and then recomputes the class centers to be the mean of the point placed in that class. Because it is a coordinate descent algorithm for (2), the sum of squares of the difference between each point and its class center is reduced by each update. This implies that the classification must eventually stabilize. The procedure terminates when the class labels stop changing.

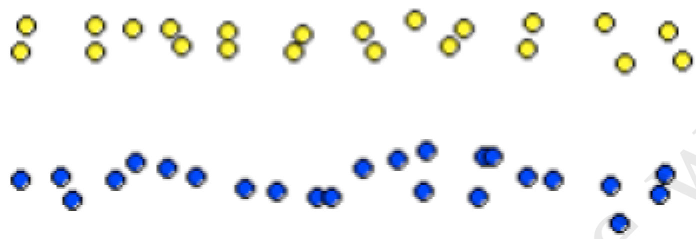
3. **Plot the direction of the first and second PCA components in the figures given.**



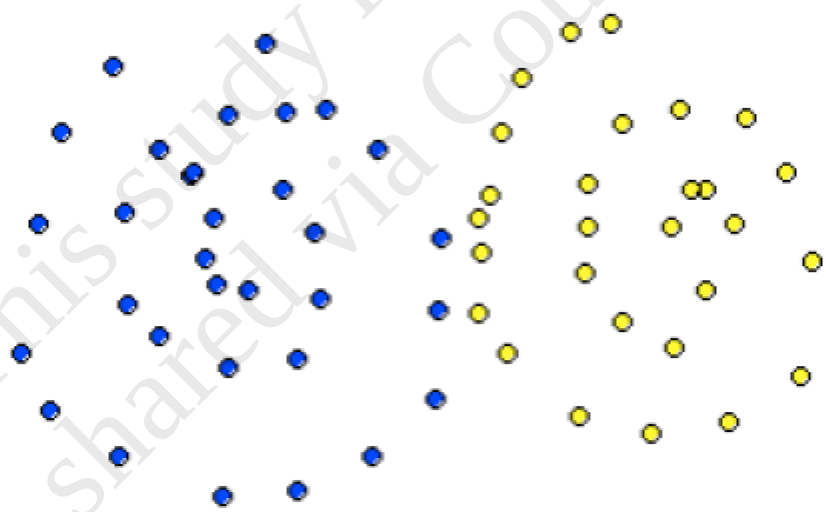
4. **Which clustering method(s) is most likely to produce the following results at k = 2? Choose the most**

likely method(s) and briefly explain why it/they will work better where others will not in at most 3 sentences.
Here are the five clustering methods you can choose from:

- **Hierarchical**
- **clustering with single link**
- **Hierarchical**
- **clustering with complete link**
- **Hierarchical**
- **clustering with average link**
- **Kmeans**
- **EM**

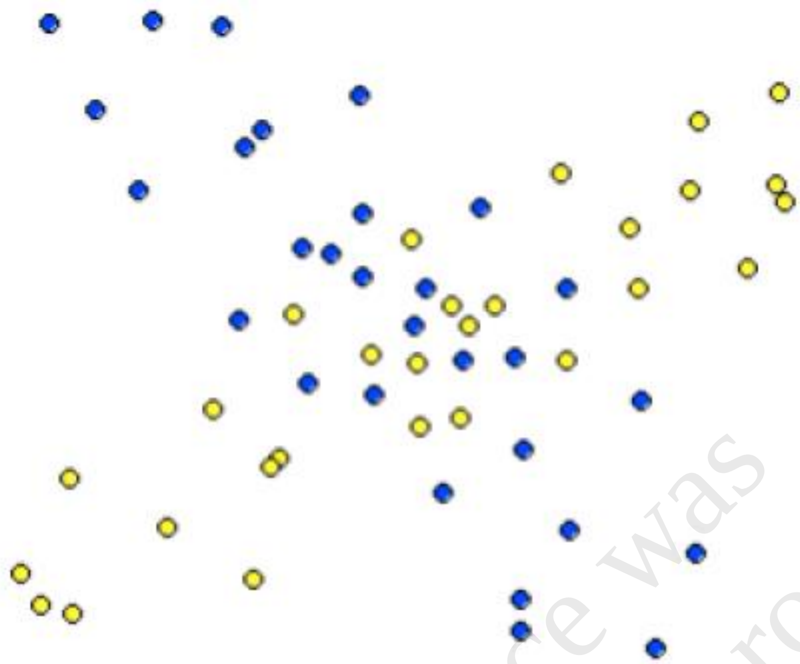


a.
Hierarchical clustering with single link is most likely to do well. EM can also produce a decision boundary that can produce such clustering result, but depending on initialization it might converge to a different set of clusters (left half vs. right half). Other hierarchical clustering's won't really work well because at some point, two intermediate clusters from different true cluster will have shorter cluster distance than two from the same true cluster.



b.

K-means or EM is most likely to do well. Hierarchical clustering wouldn't work since the early few steps will group instances near the decision boundary (note some of them are very close).



c.
Among the five methods, only EM has the capability of handling overlapping clusters. So EM is the only method that would result in such clusters.

5. Consider the following simple gridworld problem. (Actions are N, S, E, W and are deterministic.) Our goal is to maximize the following reward:
● 10 for the transition from state 6 to G
● 10 for the transition from state 8 to G
● 0 for all other transitions

S	**2**	**3**
4	**5**	**6**
7	**8**	**G**

Using the following equation to compute V(s)
V*(s) = max_a Sum(T (s, a, s_prime) [R (s, a, s_prime) + gamma * V*(s_prime)]
Given
State 6 -> G = 10
State 8 -> G = 10
All other states are 0

The actions are (N, S, E, W)
At **Iteration = 0,** the reward is

0	**0**	**0**
0	**0**	**0**
0	**0**	**0**

At **Iteration = 1,** the reward is
V(s=6) = R (6, S, G) + 0.8 V*(G) = 10 + 0.8 * 0 = 10
V(s=8) = R (8, E, G) + 0.8 V*(G) = 10 + 0.8 * 0 = 10

0	**0**	**0**
0	**0**	**10**
0	**10**	**0**

At **Iteration = 2**, the reward is
V(s=3) = R (3, S, 6) + 0.8 V*(6) = 0 + 0.8 * 10 = 8
V(s=5) = R (5, E, 6) + 0.8 V*(6) = 0 + 0.8 * 10 = 8
V(s=7) = R (7, E, 8) + 0.8 V*(8) = 0 + 0.8 * 10 = 8

0	**0**	**8**
0	**8**	**10**
8	**10**	**0**

At **Iteration = 3**, the reward is
V(s=2) = R (2, E, 3) + 0.8 V*(3) = 0 + 0.8 * 8 = 6.4
V(s=4) = R (4, S, 7) + 0.8 V*(7) = 0 + 0.8 * 8 = 6.4

0	**6.4**	**8**
6.4	**8**	**10**
8	**10**	**0**

At **Iteration = 4**. The reward is
V(S) = R (S, E, 2) + 0.8 V*(2) = 0 + 0.8 * 6.4= 5.12

5.12	**6.4**	**8**
6.4	**8**	**10**
8	**10**	**0**

6. Find a Nash Equilibrium in each case. The rows denote strategies for Player 1 and columns denote strategies for Player 2.

	A	**B**
A	**2,1**	**0,0**
B	**0,0**	**1,2**

In this case, there are two pure-strategy Nash Equilibrium at (A, A) and (B, B) and a mixed strategy equilibrium where Player 1 plays 2/3 A + 1/3 B and Player 2 plays 1/3 A + 2/3 B.

	A	**B**
A	**2,1**	**1,2**
B	**1,2**	**2,1**

In this case, there is no Nash equilibrium.

	L	**R**
T	**2,2**	**0,0**

B	**0,0**	**1,1**

In this case, there are two pure strategy Nash Equilibrium at (T, L) and (B, R) and a mixed strategy equilibrium where Player plays ½ L + ½ R and Player 2 plays ½ T + ½ B

7. You receive the following letter
Dear Friend,
Some time ago, I bought this old house, but found it to be haunted by ghostly sardonic laughter. As a result it is hardly habitable. There is hope, however, for by actual testing I have found that this haunting is subject to certain laws, obscure but infallible, and that the laughter can be affected by my playing the organ or burning incense.

In each minute, the laughter occurs or not, it shows no degree. What it will do during the ensuing minute depends, in the following exact way, on what has been happening during the preceding minute:

Whenever there is laughter, it will continue in the succeeding minute unless I play the organ, in which case it will stop. But continuing to play the organ does not keep the house quiet. I notice, however, that whenever I burn incense when the house is quiet and do not play the organ it remains quiet for the next minute.

At this minute of writing, the laughter is going on. Please tell me what manipulations of incense and organ
I should make to get that house quiet, and to keep it so.

Sincerely,
At Wit's End

a. Formulate this problem as an MDP. (For the sake of uniformity, formulate it as a continuing discounted problem, with gamma = 0.9. Let the reward be +1 on any transition into the silent state, and 1 on any transition into the laughing state.) Explicitly give the state set, action sets, state transition, and reward function.

We make the following decisions in formulating this problem as an MDP:

State set: {L, Q}, where L indicates that there is laughter in the room, and Q indicates that the room is quiet.

Action set: {O ∧ I, O ∧ ¬I, ¬O ∧ I, ¬O ∧ ¬I}, where O corresponds to playing the organ, and I corresponds to burning incense.

Assuming deterministic state transitions and rewards based upon current state and action, we have the following 4-tuples (current state, action, next state, reward) which represent correct state transitions and rewards

(a) (L, O ∧ I, Q, +1)
(b) (L, O ∧ ¬I, L, −1)
(c) (L, ¬O ∧ I, Q, +1)

(d) (L, ¬O ∧ ¬I, L, −1)
(e) (Q, O ∧ I, Q, +1)
(f) (Q, O ∧ ¬I, L, −1)
(g) (Q, ¬O ∧ I, Q, +1)
(h) (Q, ¬O ∧ ¬I, L, −1)

b. Start with policy pi (laughing) = pi (silent) = (incense, no organ). Perform a couple of steps of policy iteration (by hand!) until you find an optimal policy. (Clearly show and label each step. If you are taking a lot of iteration, stop and reconsider your formulation!)

c. Do a couple of steps of value iteration as well.

d. What are the resulting optimal stateaction values for all stateaction pairs?

(a) (L, O ∧ I, Q, +1)
(d) (L, ¬O ∧ ¬I, L, −1)
(f) (Q, O ∧ ¬I, L, −1)
(g) (Q, ¬O ∧ I, Q, +1)
(h) (Q, ¬O ∧ ¬I, L, −1)

e. What is your advice to "At Wit's End"?

If there is laughter, play the organ; if room is quite, do not play the organ and burn incense.

References:

https://github.com/hexgnu/problem_set_2
http://ttic.uchicago.edu/~dmcallester/ttic101-07/lectures/em/em.pdf
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4433949/
https://kunuk.wordpress.com/2010/09/24/q-learning/
http://www.cs.cmu.edu/~tom/10601_fall2012/hw/hw4sol.pdf
https://www.cse.iitm.ac.in/~ravi/courses/Reinforcement%20Learning_files/Solution4.pdf
https://www.coursehero.com/sitemap/schools/47-Georgia-Tech/courses/671312-CS7641/
https://oyc.yale.edu/sites/default/files/mixed_strategies_handout_0_0.pdf